Assignment Report on Burp Suite

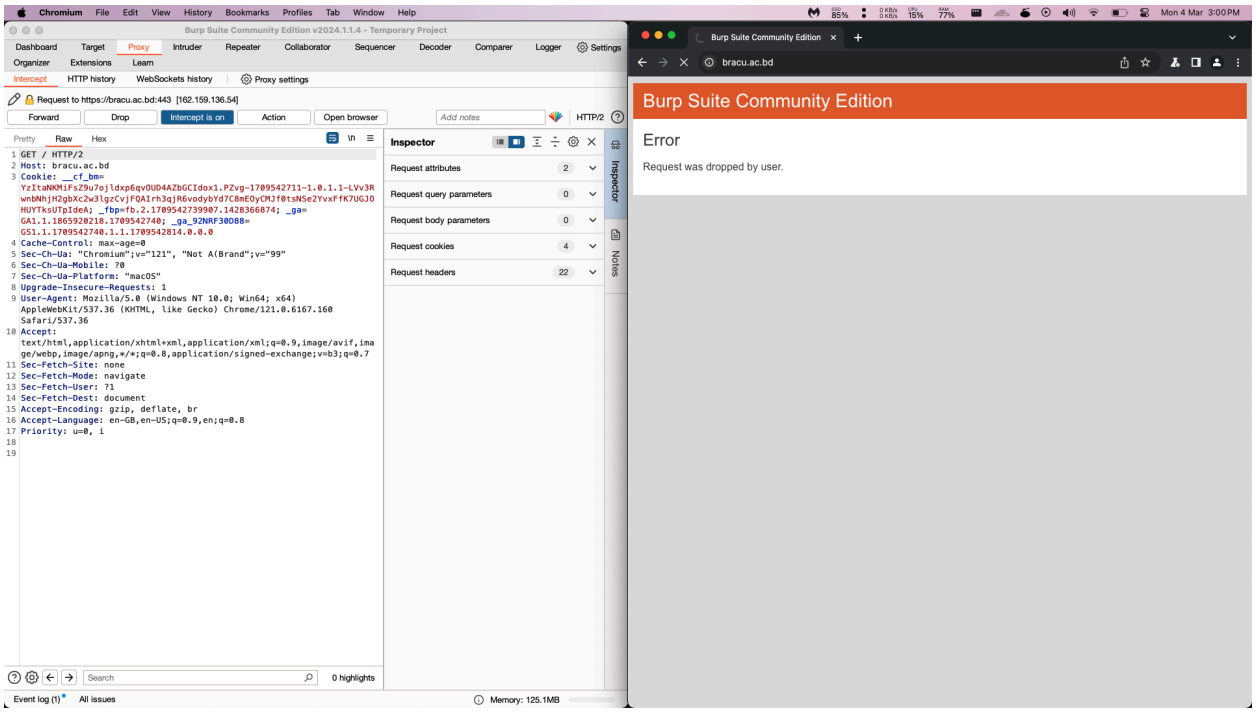
Burp Suite is a software that is used for the security testing of websites. It has several features such as proxy, intruder, repeater, sequencer, decoder, comparer, extender, etc. A report of these features of the Burp Suite software is as follows:

Proxy:

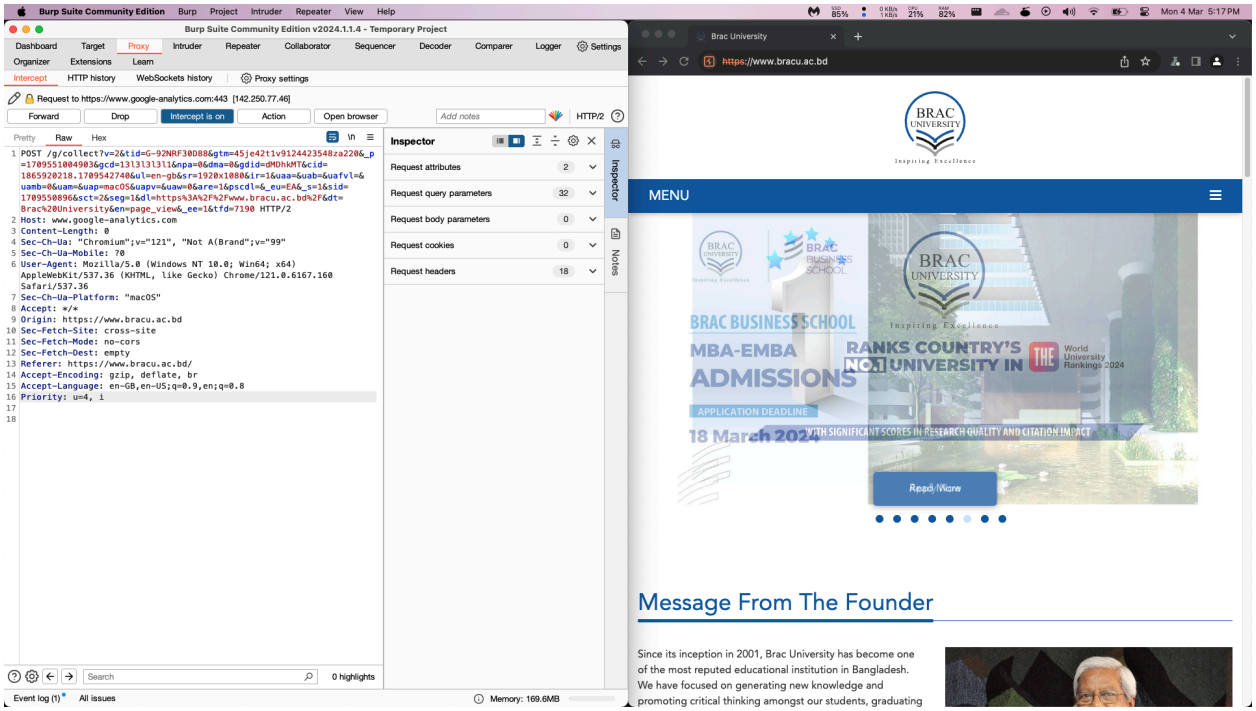
One of the best use cases of Burp Suite is that we can use it as a proxy server. As known, proxy servers are a gateway between a server and a browser. Normally, for testing the security of any website, the server-side or backend part of the application is vulnerable. When enabled, the Proxy tool pauses the traffic, displaying the requests and responses in the Burp Suite interface, where they can be edited before forwarding them to the server or browser.

a. **Intercept:**

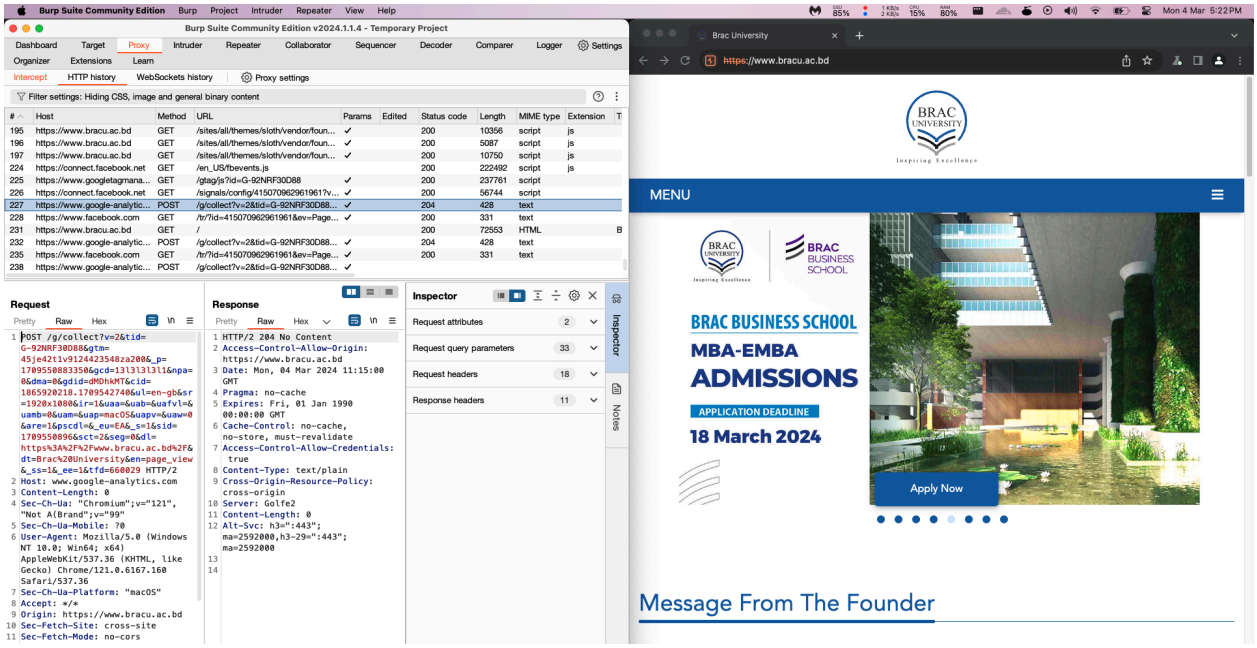
By using Burp Suite as a proxy server we can intercept website connection to the browser of our choosing. By default, a Chromium browser with the Burp suite will open, but we can connect it to the default browser we normally use. To open the browser we have to the Proxy tab of Burp Suite click the intercept is on button then enable the browser to open up. Next, we have to type the name of a website, for our example, we will type bracu.ac.bd. Now, we see the below screen where the browser is in a constant loading stage.



Here, the burp suite software intercepts the incoming and outgoing connection between the browser and the website. If we click forward once, it won't show any changes on the website but if look at the raw tab opened in the Burp suite, it will show some changes in the requests. With every requests, there are changes in the raw tab, where we can see the Request type, hostname, cookies etc. This explains how many requests are working behind the screen when we go inside a website. After clicking the forward button a few times, we can finally see the website loading properly.

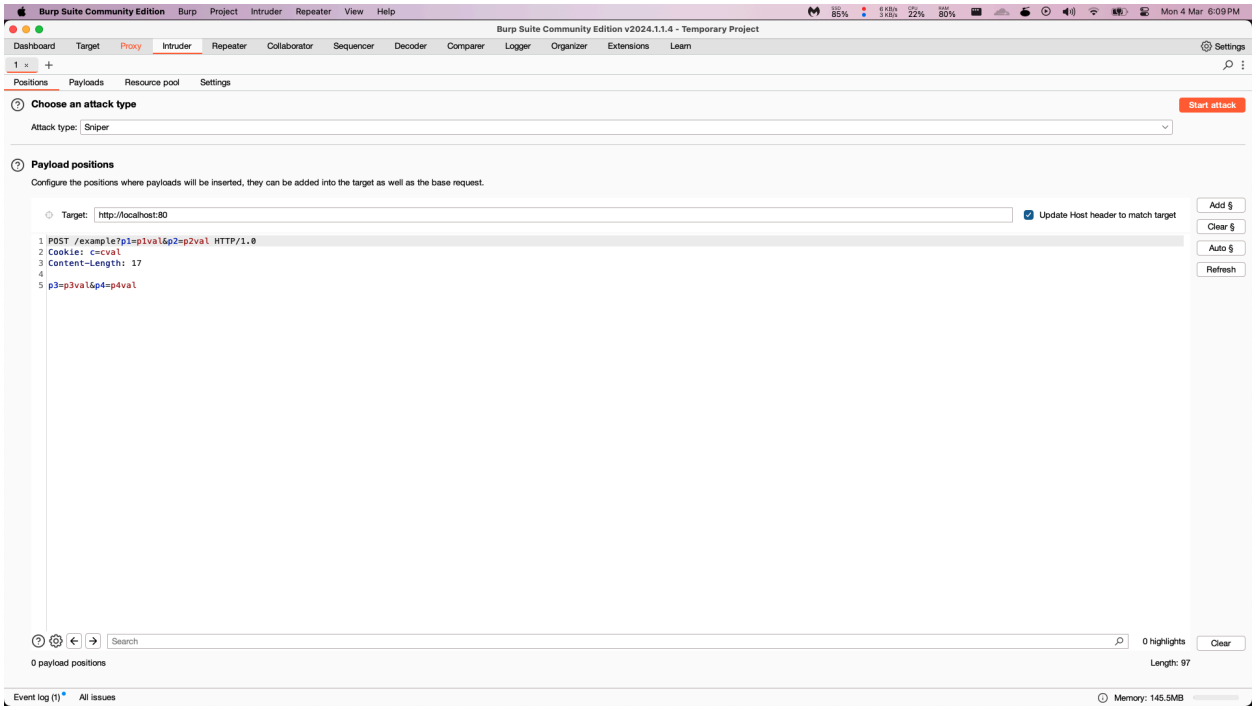


We can also check the HTTP request history between the proxy tab and browser that was created while trying to go to the websites. If we click one of the requests, we can check the response request as well as inspect several parts of the requests such as headers, cookies, etc.



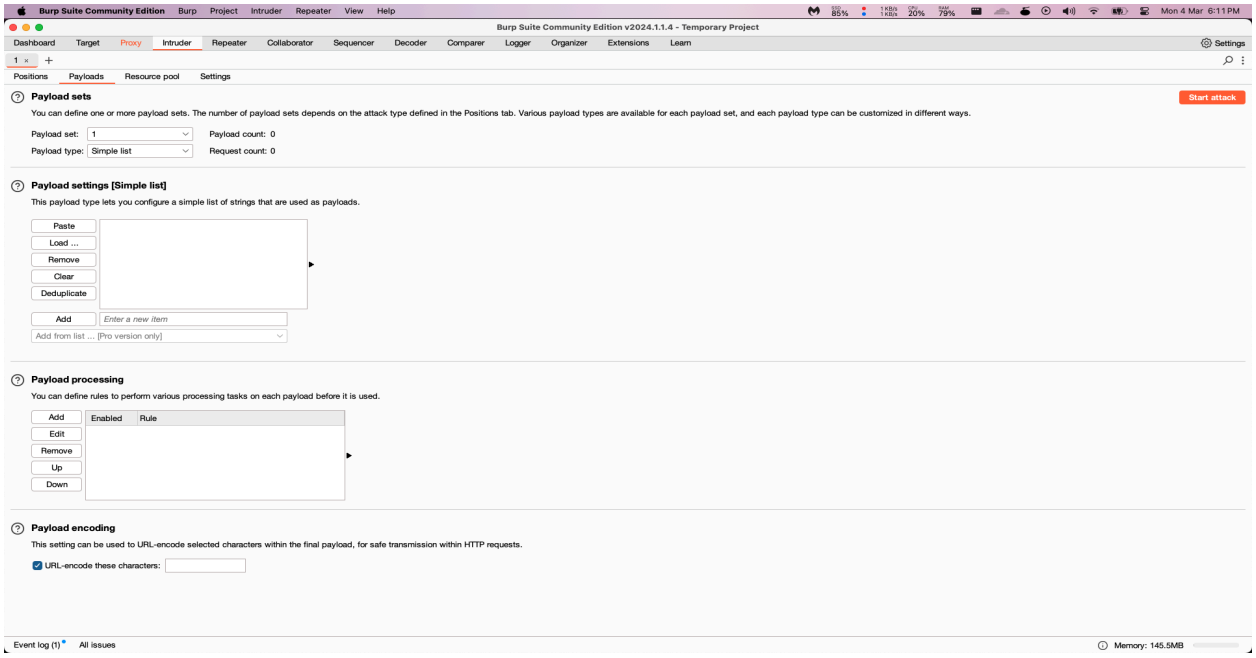
This intercept feature is crucial for security testing, allowing detailed inspection and manipulation of messages to identify vulnerabilities or understand the behavior behind every incoming and outgoing request to the website.

Intruder: Another important functionality of Burp suite software is to act like an intruder for logging into any website. To use this function we need to use payloads, where a lot of common passwords are already generated. When we cannot attack using manipulation or changing codes, and is useful for such as brute-forcing passwords, testing input validation, SQL Injection, or XSS. Here we have to first connect the intruder with any of the requests or connections. For this example, let's check the post request of the given example inside the intruder tab. We won't be attacking any website but will only check how things work. We will see the following screen after going to the intruder tab:



We can only execute the payloads where there are input options. On the right part, we can see the add, clear, auto-select, and refresh options. This would change the input fields respectively. The software selects the input fields automatically by default.

Besides position, we have to option to determine the payloads that will be used in the input part. If we into that tab there are several selective options to choose from as to how many sets of payloads we have, the type of payloads where there are several types to choose from.

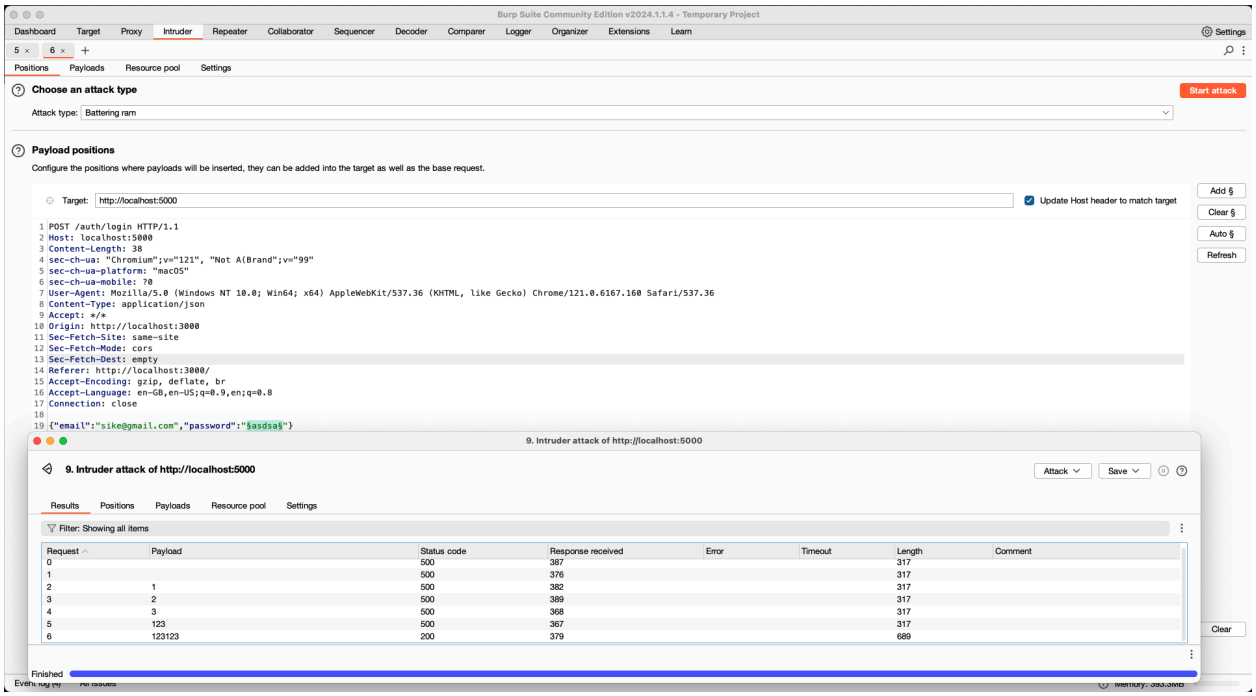


We can change the payload settings and also the processing of payloads. For every type of payload, the payload settings will change the requirement as each type has different requirements. We can determine or set rules as to how the payloads will process themselves when the attack takes place. We can change the settings of the intruder attack in different ways such as how many attacks will take place in one second, how many iterations will be done in each second, and specific attack-based settings to properly build the attack.

There are several ways to attack using the intruder, these are explained as follows:

a. **Sniper:** In this attack, we can only use one set of payloads, but we can position the payloads to more than one input. Firstly, the payload will be selected for the first input position and after that for every input position, and so on.

b. **Battering Ram:** For this attack, we can use only one input. The ideal use case for this attack is only attacking to determine the password. This attack iterates all the payload strings in one input which is useful for testing multiple parameters with the same value.



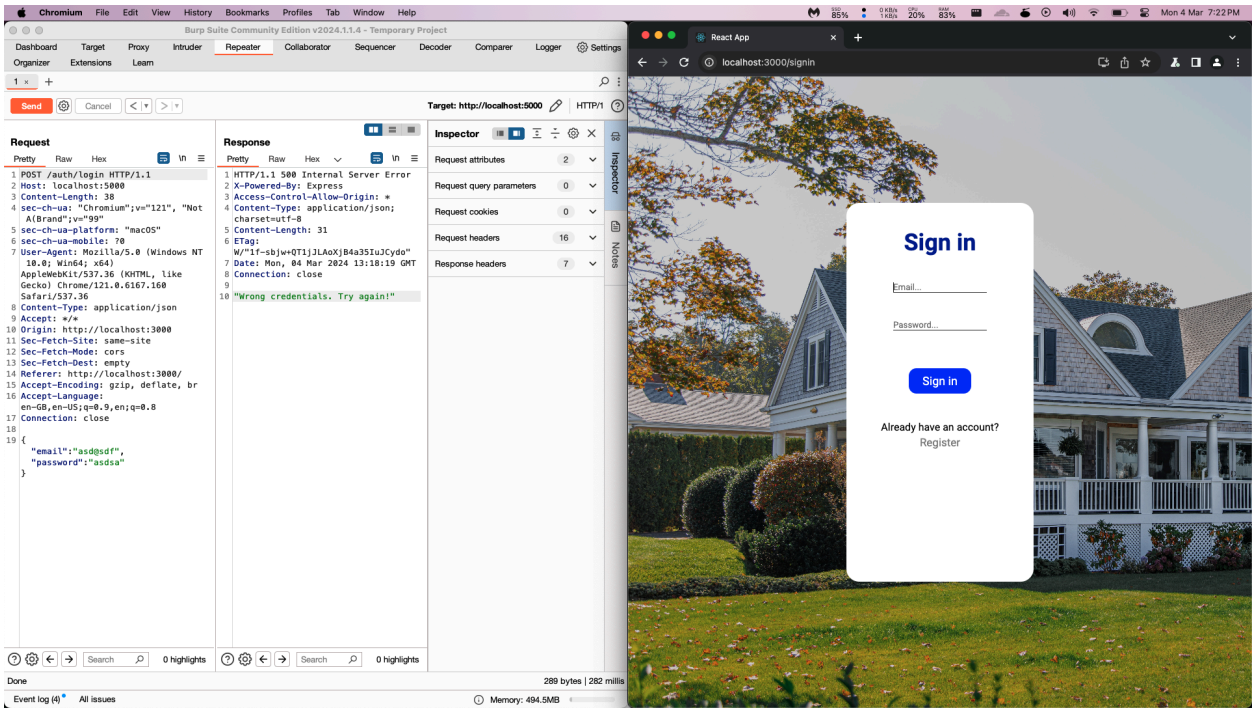
In the above image, we have used battering ram to attack a locally hosted site with different passwords that we have given in the payload. In the last attack, we are seeing the status code 200 OK meaning that our attack was successful.

c. **Pitchfork:** It uses more than one payload set, where we can use different payloads with different inputs. Then iterates through multiple payload sets simultaneously, assigning one payload from each set to each input position. For example, if we want to attack

using a username and email and with different sets of payloads, we can use Pitchfork for the attack.

d. **Cluster Bomb:** This is most strongest attack on intruders making it the most exhaustive and resource-intensive attack compared to others. It generates a combination of payloads across all input positions, and then each combination of payloads can be used. For example, if we have 100 usernames and 100 passwords, there will be 100 * 100 attack combinations.

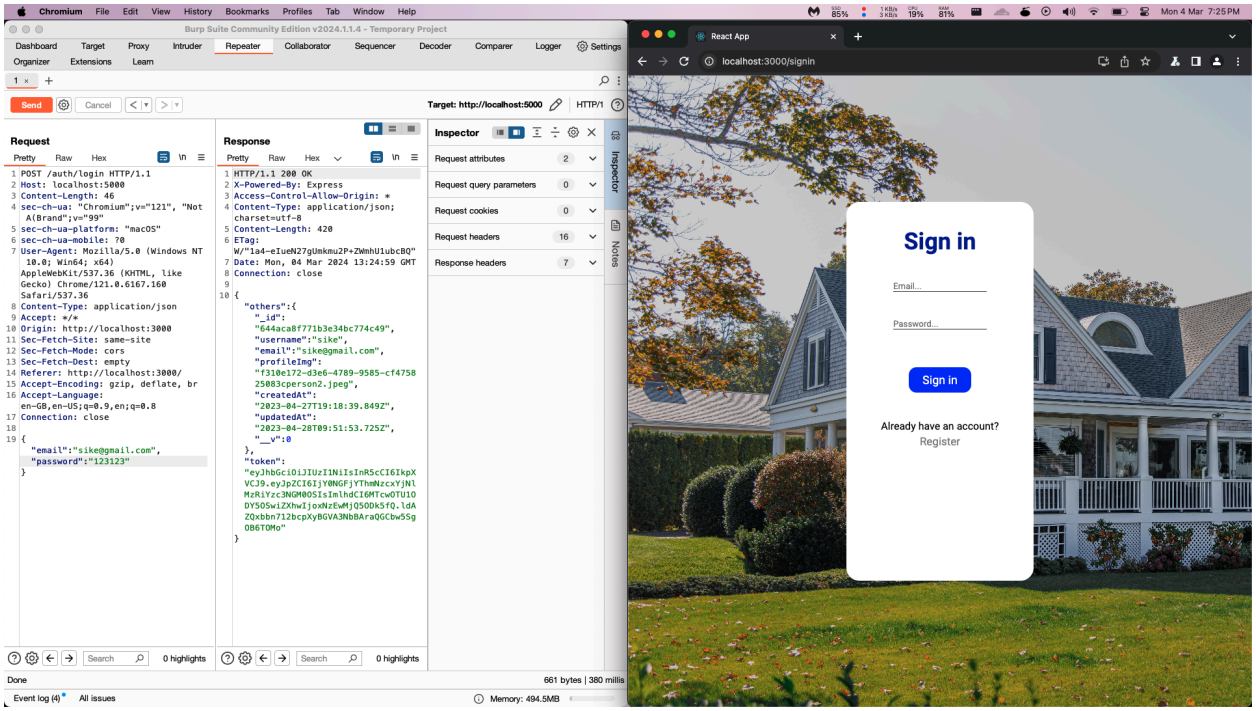
Repeater: With this functionality, we can manually test individual requests. We can also modify requests and resend them to observe the behavior of the application or to exploit vulnerabilities. Similar to the intruder, we have to send which request we want to use in the repeater by right-clicking from anywhere in the Burp Suite software. If we see an example of trying to log in to hosted locally site we will see the below image:



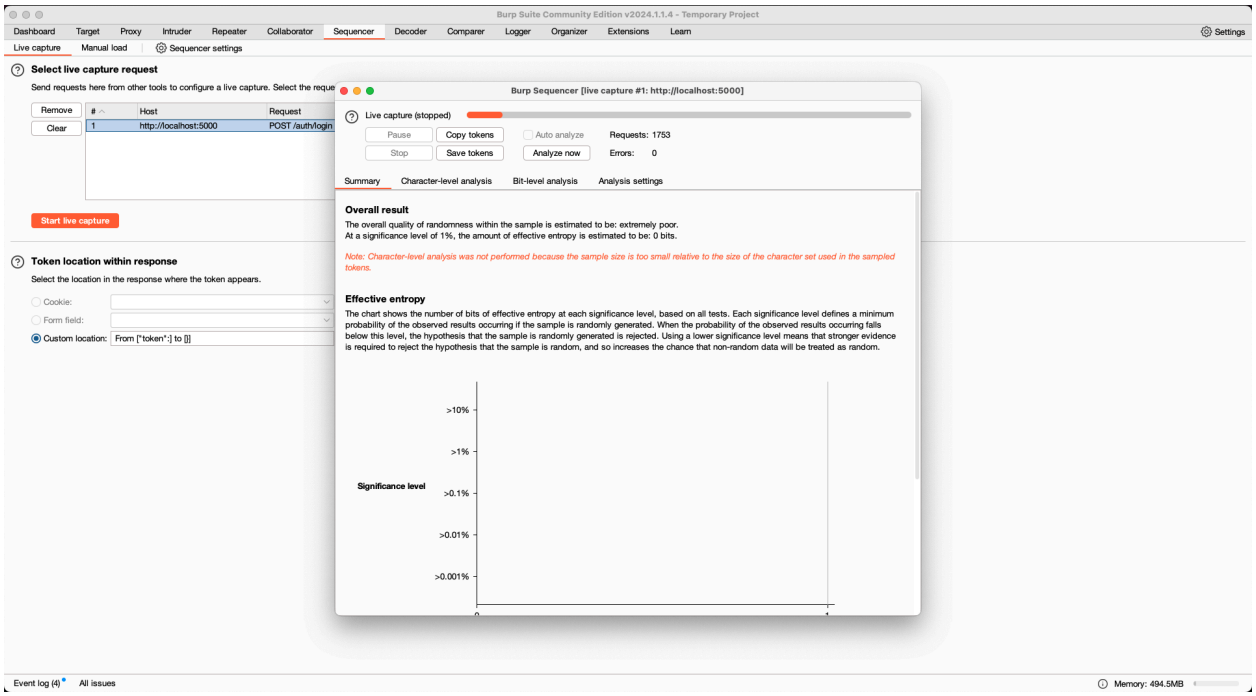
In this image, we see the POST request to the website and at the left side of the burp suite. To understand what we are trying to achieve I have also opened the login page of the locally hosted side beside the software. Initially, we tried a false username and password which cannot be used for login. After sending the request using Burp Suite, we get a response that we can see beside the request tab. As explained, due to having wrong credentials, we are seeing the error message and bad request alert in the response tab.

Again if you check the below image, for the same website we have used the correct credentials this time. Now if we send the request using Burp Suite then get proper logged-in response. If we see the response tab, we have generated a JSON web token so it proves that the credentials

were right. Moreover, similar to the proxy we can also check the request header, cookies, etc. to successfully identify any sort of vulnerabilities in the website.

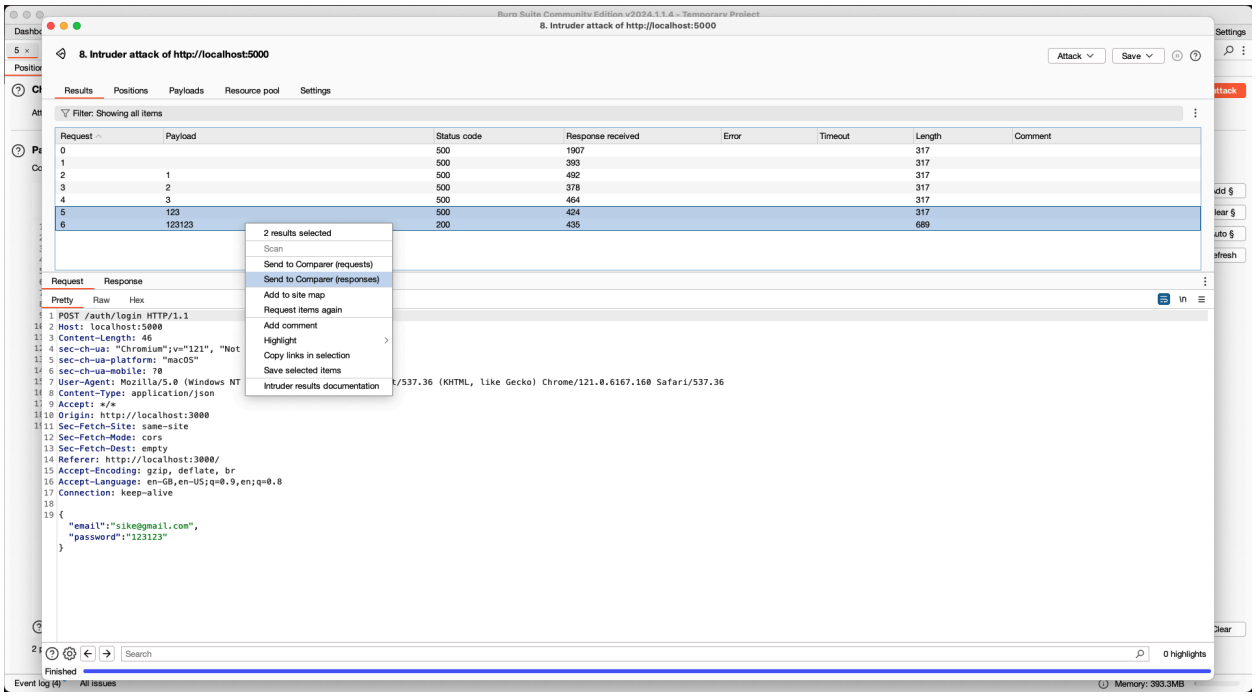


Sequencer: Sequencer is used to scrutinize any sort of random session tokens and important data on the website. If we see the above generated JSON web token, it is generated randomly for the user and will expire after some time. It must be random every time because, if it is specific, then all the data of the user can be stolen if found out or if a brute force attack is used. So, in every website when we are logging in a new session ID is created randomly. The Sequencer is a great tool for identifying the randomness or pseudo-randomness of this session and other important data in a website. The sequencer performs several different mathematical evaluations against several pseudo-random numbers in an attempt to determine the sources of entropy from when they were generated. For this, live capture can be used to generate sample values by issuing a crafted request that will result in new values being assigned. This is often done by removing an existing cookie value from a request so that the response provides a new session token in the form of a new cookie response header. For using the sequencer, we have to first send the request to the repeater, after that, we have to set the parameters of cookies or any headers where there random or pseudo-randomly generated strings. After choosing a header, we can then send it to the sequencer and start live capturing to see the result. For our example, we have selected the JSON web token that we have seen previously. We need to add a custom configuration option to select it. After properly choosing the option we can then start live capture to analyze the randomness of the token. We can the work happening in the below image.

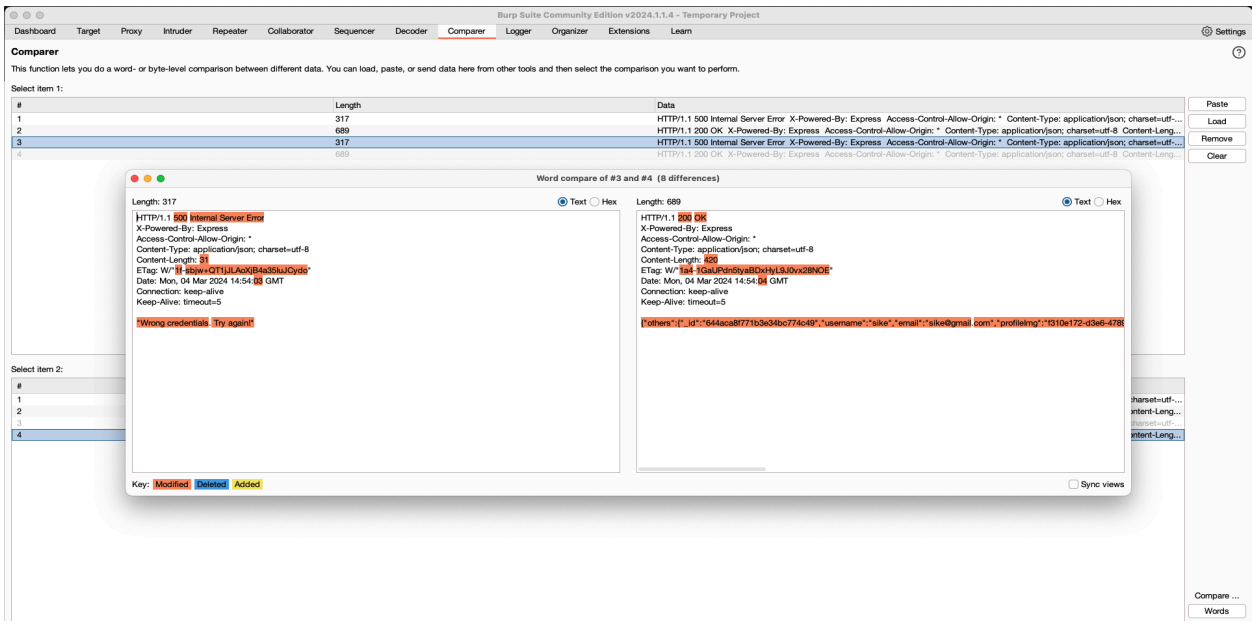


Decoder: As the name suggests, we can use a decoder to decrypt any sort of encrypted numbers from one type to another type of encryption. It supports plain text, URL, HTML, ASCII index, Base64, Hex, Octal, Binary, and Gzip formats. To use this we first to write something in the textbox. For example, if try to encrypt a plain text file we are going to “CSE496: Ethical Hacking”, then let’s try the base64 encryption. After choosing it, the encryption will occur automatically. Decoder also lets us do more than one level of encryption, we can convert the coded base64 string to ASCII index format. Then let us do one more level by choosing binary format. So, after that, we got a string which is encrypted three times. We know how to decrypt it but for others, it will be really hard to determine the exact formats the message was decoded.

Comparer: It is used to compare different responses or requests with each other to determine any vulnerabilities. If we see an example of the previous login page of, if we extend the previously used case of battering ram attack, let's choose the last two attacks having different status codes.

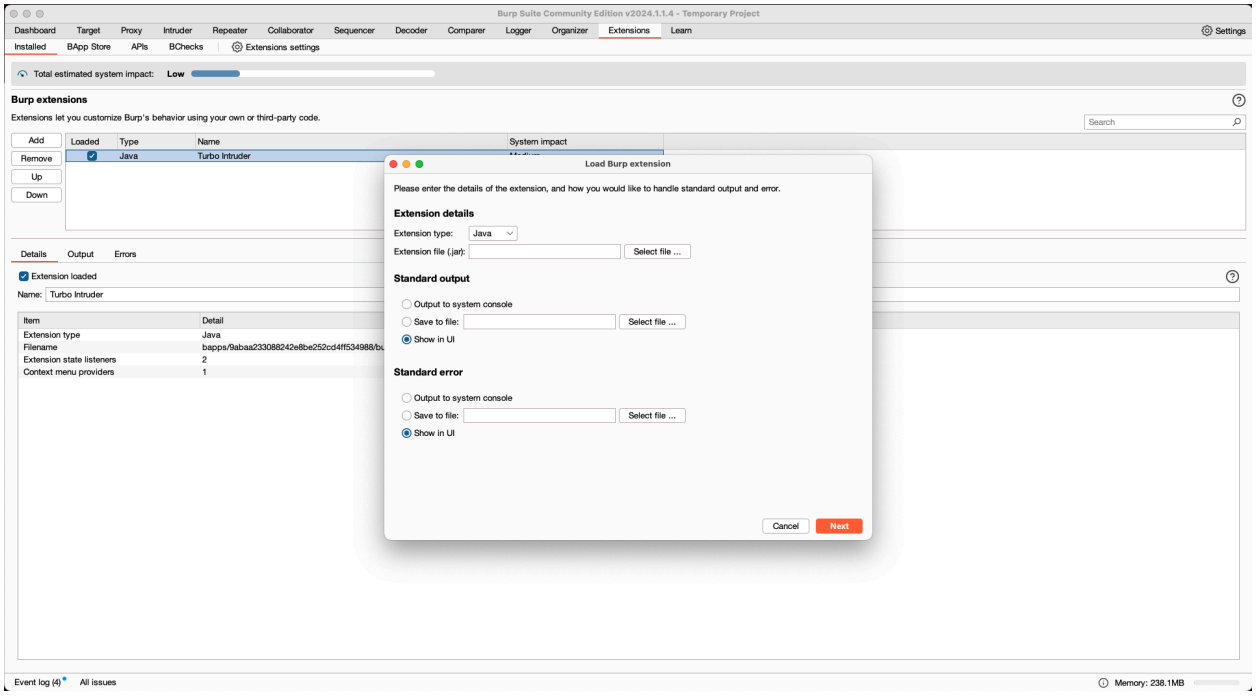


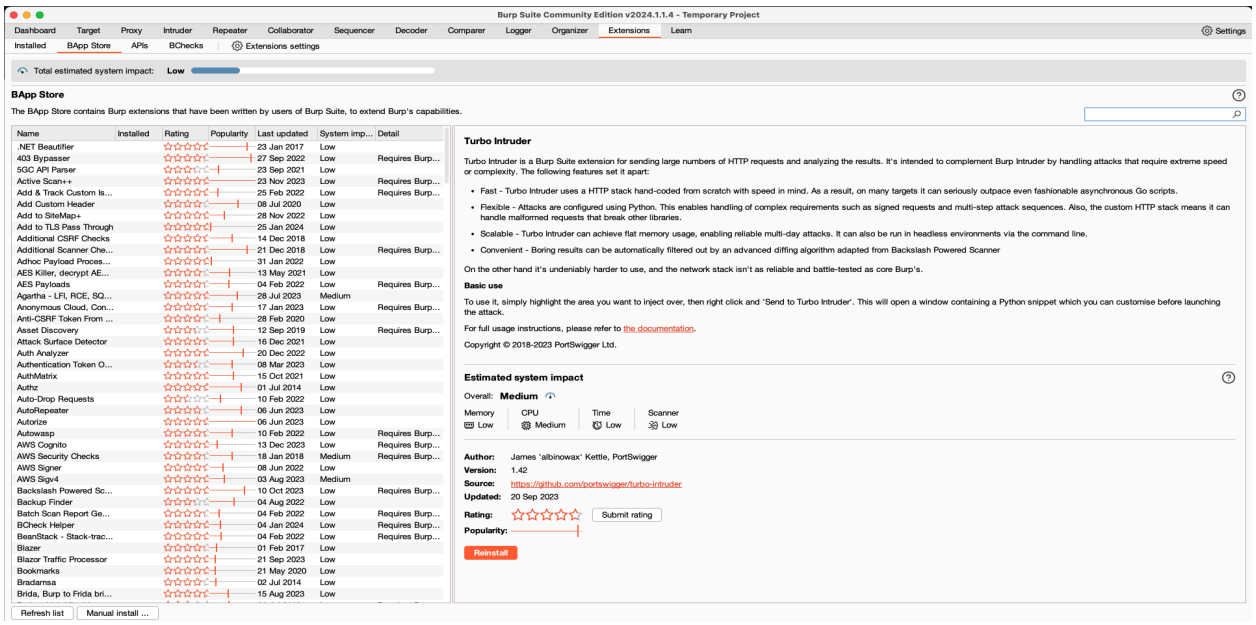
After we send this to the comparer (responses) we have we will see both of these requests inside the comparer. We have to click on the word option in the bottom-right corner for seeing the comparison and differences between the two requests.



As we can see, comparer has highlighted all the differences between the requests. In this way, we can determine any sort of data to identify differences between responses and requests.

BURP Extensions: Extensions can be used to enhance the current usability of the vanilla burp suite software. In the extensions tab, we can manually install an extension using the add button, which will prompt us to add extension details. The extension type can be of Java, Python, or Ruby programming languages, and the extension file will have .jar extension.





If we do not want to install manually, we also have the option to download any Burp extension from the BApp store automatically. All legal extensions for the BURP suite can be found in this tab right beside the installed tab on the top right. After installing the extension we can see that applied in the installed tab. Extensions can also be edited or removed as needed.